

Pascal

```
procedure Zamien (var a, b: integer);  
var  
    t: integer;  
begin  
    t := a;  
    a := b;  
    b := t;  
end;
```

C++

```
void Zamien(int &a, int &b)  
{  
    int t=a;  
    a=b;  
    b=t;  
}
```

Procedurę należy wywołać z parametrami aktualnymi $a[i]$ i $a[k]$: `Zamien(a[i],a[k]);`



Ćwiczenie 2.

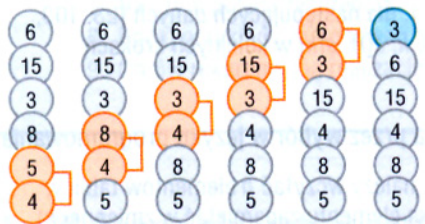
Korzystając z przykładów 1. i 2, napisz program realizujący algorytm sortowania przez wybór zbioru składającego się z n liczb całkowitych, wprowadzanych z klawiatury i wczytywanych do tablicy.

1.2. Sortowanie bąbelkowe



Sortowanie bąbelkowe ciągu n liczb polega na porównywaniu parami kolejnych liczb i przestawianiu ich, jeśli występują w niewłaściwej kolejności.

Aby uporządkować ciąg liczb (np. rosnąco) przy zastosowaniu algorytmu sortowania bąbelkowego, porównujemy liczby parami. W przykładzie widocznym na rysunku 1. porównywanie przebiega od dołu ku górze. Jeśli ostatnia liczba (umieszczona na samym dole) jest mniejsza od przedostatniej (drugiej od dołu), to zamieniamy je miejscami.



Krok I Krok II Krok III Krok IV Krok V Krok VI

Rys. 1. Przykładowy (pierwszy) cykl sortowania bąbelkowego

Następnie porównujemy liczbę przedostatnią z liczbą trzecią od dołu – w przypadku gdy przedostatnia liczba jest mniejsza, zamieniamy je miejscami. Czynności powtarzamy.

Nietrudno zauważyć, że już w pierwszym cyklu sortowania liczba najmniejsza (w tym przypadku liczba 3) zostaje ustawiona na właściwym miejscu. Aby ustawić we właściwej kolejności pozostałe liczby, należy powtarzać cykl sortowania, dopóki nie zostaną wykonane wszystkie przestawienia.



Przykład 3. Algorytm sortowania bąbelkowego w porządku rosnącym

Zadanie: Uporządkuj rosnąco n liczb, stosując algorytm sortowania bąbelkowego.

Dane: Tablica liczb $a[1..n]$.

Wynik: Tablica liczb $a[1..n]$, w której liczby ułożone są w porządku rosnącym.

Lista kroków:

1. Zaczynj algorytm.
2. Zmiennej *poczatek* przypisz wartość 1: $poczatek := 1$.
3. Jeśli $poczatek \geq n$, przejdź do kroku 9.
4. Zmiennej i przypisz wartość zmiennej *poczatek*: $i := poczatek$.